MAY1735

ISU Department of ECprE

Dr. Daniels

Kyle Fischer - Team Lead

Michael Linthicum - Communications Lead

Daniel Shauger - Concept Holder Lead

Sam Neff - Webmaster

Nick Juelsgaard - Schedule and Planning Lead

may1735@iastate.edu

http://may1735.sd.ece.iastate.edu/

Revised: 10/15/16

CyMote for CprE 185

Project plan V1.0

Contents

[1 Introduction 2](#_Toc464164451)

[1.1 Project statement 2](#_Toc464164452)

[1.2 Purpose 2](#_Toc464164453)

[1.3 Goals 2](#_Toc464164454)

[2 Deliverables 3](#_Toc464164455)

[2.1 CyMote (beta model) 3](#_Toc464164456)

[2.2 PC Wrapper 3](#_Toc464164457)

[3 Design 4](#_Toc464164458)

[3.1 Previous work/literature 4](#_Toc464164459)

[3.2 Proposed System Block diagram 4](#_Toc464164460)

[3.3 High Level Wiring Diagram 5](#_Toc464164461)

[3.4 Validation 5](#_Toc464164462)

[3.5 System Design 6](#_Toc464164463)

[3.6 Detail Design 6](#_Toc464164464)

[3.6.1 Input/Output Specification 6](#_Toc464164465)

[3.6.2 User Interface Specification 6](#_Toc464164466)

[3.6.3 Hardware/Software Specification 6](#_Toc464164467)

[3.6.4 Prototyping Specification 6](#_Toc464164468)

[4 Project Requirements/Specifications 7](#_Toc464164469)

[4.1 functional 7](#_Toc464164470)

[4.2 Non-functional 7](#_Toc464164471)

[5 Challenges 7](#_Toc464164472)

[6 Timeline 8](#_Toc464164473)

[6.1 Phase 1 – Alpha Model/Planning 8](#_Toc464164474)

[6.2 Phase 2 – Pinciple Project 8](#_Toc464164475)

[6.2.1 Phase 2a – Beta Model 9](#_Toc464164476)

[6.2.2 Phase 2b – Project Code 9](#_Toc464164477)

[6.2.3 Phase 2c – Presentation 9](#_Toc464164478)

[7 Conclusions 9](#_Toc464164479)

[8 References 9](#_Toc464164480)

[9 Appendices 10](#_Toc464164481)

# 1 Introduction

## 1.1 Project statement

CprE 185 and EE 285 use a device called the Arduino Eslpora. It is a handheld controller with a mix of I/O on board, including some buttons, a joystick, Bluetooth, and a three dimensional accelerometer. These classes use this device to obtain real-world data and create interesting programming challenges for the students. Unfortunately, this device is now obsolete.

This senior design project will attempt to create a new platform for these classes to allow for the interesting challenges to continue for these classes. This new device, dubbed the CyMote, will attempt to mimic the important functions of the Esplora and be easily accessible for the new engineering students.

## 1.2 Purpose

The declared purpose of both CprE 185 and EE 285 emphasize a goal of using programming to solve engineering problems. It is important for students to associate the work they do in class to solving problems in the real world. The Esplora is that connection. It allows both user input in the form of the joystick and buttons, but it also allows for collection of the effect of the laws of physics through the accelerometer.

We want to develop a device that will allow students a meaningful pathway into solving real-world problems. This device will reliably provide the aforementioned I/O, withstand constant handling, and be easy enough for a freshman to figure out. The three main tenants of the project design are:

1. Reliability
2. robustness
3. Simplicity for End user

## 1.3 Goals

The goal of the project is to create a demonstrably robust CyMote device that will be capable of all required functions. As students, our goal is to learn and grow as technically skilled engineers. As future professionals, our goal is to refine our skills in communication, teamwork, presentation, as well as project and time management. As ambitious, tech savvy young people, our goal is to complete a project that excites both us and other people. As seniors, our goal is to graduate and GTFO.

# 2 Deliverables

There are two main deliverable items for this project: (1) A physical beta model of the CyMote for display and presentation, and (2) a PC wrapper that will provide a student with information from the CyMote for use in class.

## 2.1 CyMote (beta model)

1. This project will deliver at minimum one beta model of the CyMote design.
2. This device will contain all necessary control, communication, and I/O hardware.
3. It will be capable of collecting data from a fall in the Atrium of Coover and transmitting it over Bluetooth.
4. Its power source will be onboard rechargeable batteries.

## 2.2 PC Wrapper

1. The PC wrapper will be on a laptop that can travel with the CyMote.
2. It will handle all communication through a Bluetooth transceiver on the laptop to the CyMote itself.
3. It will transmit CSV values of the following through a cmd line to an output of the user’s choice:
	1. Accelerometer values (x,y,z)
	2. Button states
	3. Joystick states
	4. LED states
4. It will be capable of recovering from common failure modes.

# 3 Design

## 3.1 Previous work/literature

A previous senior design team attempted the same project but as not successful. There was no software written that we found or used. The previous team acquired parts and managed to assemble a nonfunctional prototype board, of which some of the peripherals were recycled for our project.

## 3.2 Proposed System Block diagram

A basic block diagram of the system as seen in Figure 1 below.

**Basic System Architecture**

**Figure 1**

## 3.3 High Level Wiring Diagram

A basic One-Line Diagram of the electrical connections.

**One-Line Diagram**

**Figure 2**

## 3.4 Validation

Formal testing is not in the scope of this project as we are not planning on having a final, deliverable PCB design.

## 3.5 System Design

The CyMote needs to be:

* Durable
* Hold a program
* Portable

## 3.6 Detail Design

\*Details in V2.0\*

### 3.6.1 Input/Output Specification

Input is taken from various sensors including a 9-degrees of freedom sensor, joystick, buttons, and an audio codec**(?).** The sensor data is then outputted in real time through USB or BLE and is displayed on a monitor in the command line.

### 3.6.2 User Interface Specification

A program is run from the command line (bash) that is supplied with various tags indicating which sensors to use. While running, the program will output the data as comma-separated values. Various LEDs will be present on the board itself to indicate if a sensor is active or dormant.

### 3.6.3 Hardware/Software Specification

The CyMote will use the ATSAMB11 to control various peripherals. The peripherals used are:

* Joystick w/ button
* 4 buttons
* Power Switch
* 9-Degrees of Freedom sensor (LSMD90)
* Audio Codec (**?)**
* 3.7V Battery
* RGB LED

The software component will be written in C using Atmel Studio and ASF project structure.

### 3.6.4 Prototyping Specification

The breadboard prototype must be durable enough to withstand a 30-foot drop. The breadboard prototype will contain all sensors connected in a fashion that will be easily portable to a PCB design.

# 4 Project Requirements/Specifications

## functional

1. Droppable – survive a 30’ drop in a soft container of some kind
2. Robust enough to have people plugging/pressing/slamming/charging often
3. Communicate with a partner PC over Bluetooth wirelessly
4. Low power consumption from onboard rechargeable batteries
5. An audio codec to take microphone input and output to a speaker?
6. I/O components
	1. 9° of Freedom – 3D accelerometer
	2. Joystick – (x,y analog inputs, button press)
	3. 4 game controller style input buttons
	4. Status LEDs for all the devices
	5. Tricolor LED for display/status/fun
7. Power on/off switch

## 4.2 Non-functional

1. Look like a game controller
2. Stream data in real time – fast enough to register current movement on screen as having no lag
3. Battery life long enough to last a full day (up to 8 hours) of use
4. Beta model built on a handheld size PCB

# 5 Challenges

We are trying to face the challenges for this project head on. One of the early challenges of this project was striking a definition of done for the device that was both realistically manageable and would still provide a useful product for the client. We spent a good amount of time talking about what features we would put in scope for sure, what features we were going to scope out, and which features we could make a better decision on later.

Another challenge we face that the device must not only function properly, it must also adhere to certain other guidelines that we have not had to face as students before. Namely, this device is the first step towards another device that will be mass-produced and used in a classroom setting. We have to keep the production model in mind when we design the alpha and beta models. We have to keep power consumption, communication-bus protocol reliability, and individual part support in mind as we pick parts and assemble the device.

# 6 Timeline

We will accomplish the project in two main phases. Phase 1 will be a planning and testing phase. We will be constructing an alpha model of the device on a breadboard and creating some simple commands to talk with the device. Phase 2 will be three parts: beta model construction, software/PC wrapper composition, and project presentation. Here is a Gantt Chart of the overall project:



An important note is that we are specifically planning not to work on any school-sanctioned break. We have removed Thanksgiving break, the winter break, and spring break from our scope for work.

## 6.1 Phase 1 – Alpha Model/Planning

Phase 1 work ends in dead week of fall semester when we present our Project Documentation. Phase 1 is comparatively flexible to Phase 2. The main objective of Phase 1 is a breadboard model of the device that we can use to simulate the actual function of the final device. The definition of done for this project is what makes this phase so flexible.

The other main objective of this project is the presentation of the project plan. We will be presenting our project as a concept during dead week of the fall semester. This includes design documentation, a project plan, and all necessary visual aids. However, in order to keep this project moving forward, we are planning on starting Phase 2 before Phase 1 is complete.

## 6.2 Phase 2 – Pinciple Project

We are planning to start this phase of the project the week after we return from Thanksgiving break. A lot of this work will actually be overlap with the Alpha model, but since we are not planning on working during winter break, we need to start development of the beta model before we leave for four weeks. The three phases of the project will also overlap, and our Gantt chart lays out only a possible optimal schedule for development.

### 6.2.1 Phase 2a – Beta Model

We will be working on the beta model as we finish the alpha model. We are still deciding whether to design a PCB for the beta model, or if we will just be building a more compact breadboard. Since this part of the project is necessary for the PC wrapper to be fully developed, we set the deadline for the week before spring break.

### 6.2.2 Phase 2b – Project Code

Again, this phase will overlap with all other phases, but in theory we will begin working on the PC wrapper code and the MCU code in earnest sometime after Thanksgiving break. Because we want to ensure our project presentation has the best chance for success, and because we are all graduating in May and know that our schedules will be very full by the end of the year, the deadline for this project is 3 weeks before our presentation. This should give us plenty of room to make mistakes or to cross the finish line a little early if possible.

### 6.2.3 Phase 2c – Presentation

Our final presentation will be in the first week of May, but we would like to have a lot of supporting visual aids, so we plan on preparing for the presentation in mid-February. With all other phases of the project winding down three weeks before our presentation, we should have enough time to create a well thought-out display.

# 7 Conclusions

This project will have two main phases and two main deliverable items. We will build a test alpha model by the end of the fall semester, and we will be able to communicate with the device on basic levels. As we present the project at the end of the fall semester, we will be waist deep in phase two, which will end with the two project deliverables: a sufficiently reliable, robust, and simple CyMote device, and a PC wrapper that allows communication with the device.

The full intention of the beta model of the device and the software is for another group to take our design in the future and develop the production model of the CyMote for use in CprE185 and EE285.

# 8 References

There are no official references right now, but our unofficial references are:

* Previous senior design group that did not quite complete the project with an Edison controller
* Atmel Studios template projects for the ATSAMB controller
* Dr. Daniels/EE185 class documentation

# 9 Appendices

Appendix will grow as this documentation becomes more formalized.